

**LAPORAN HASIL PRAKTIKUM
PEMROGRAMAN WEB & MOBILE I**



NAMA : HUSSAIN SYACH NURALAM
NIM : 193020503029
KELAS : A
**MODUL : ASYNCHRONOUS JAVASCRIPT AND
XML (AJAX)**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2021**

BAB I

PENDAHULUAN

1.1 Tujuan

Tujuan dari dilaksanakannya praktikum Pemrograman Web dan Mobile, modul IV tentang Asynchronous JavaScript and XML (AJAX) adalah sebagai berikut.

- 1.1.1) Mahasiswa mampu membuat program dengan menggabungkan fitur PHP dan JavaScript.
- 1.1.2) Mahasiswa mampu membuat program web yang dinamis.

1.2 Landasan Teori

AJAX atau Asynchronous JavaScript and XML adalah teknik yang digunakan untuk membuat website yang dinamis. Artinya website mampu mengupdate dan menampilkan data baru dari server tanpa perlu melakukan reload. Salah satu contoh penggunaannya misalnya pada update jumlah angka likes dan komentar pada media sosial Instagram, Facebook, Twitter, dan lainnya.(Aprilia, 2021)

Sesuai namanya, AJAX terdiri dari JavaScript dan XML yang bekerja bersama. JavaScript adalah bahasa pemrograman untuk mengelola konten website yang dinamis. Sementara XML (eXtensible Markup Language) digunakan untuk memuat dan membawa data dari server ke browser.

AJAX JavaScript dan XML ini bekerja secara asynchronous untuk berkomunikasi dengan server. Proses pertukaran informasi ini dilakukan di background. Artinya, saat AJAX JavaScript dan XML bekerja, halaman dapat tetap diakses oleh pengunjung website.

AJAX sangat berfungsi dalam pemakaiannya dalam program website. Fungsi dari AJAX adalah:

1.2.1) Mengirim dan Mengambil Data dari Server

AJAX dapat digunakan untuk mengirim pesan ke server lalu mengambil hasil data dari server ke browser. Prinsip yang dikerjakan

pun adalah asynchronous. Jadi, selama proses mengirim pesan terjadi, browser bisa tetap terus digunakan sambil menunggu respon dari server.

1.2.2) Mengupdate Tampilan Website Tanpa Harus Reload

Dengan adanya AJAX, pengunjung website bisa lebih nyaman mengakses website tanpa perlu berulang kali reload halaman. Hal ini terjadi karena AJAX hanya mengirimkan sebagian data yang dibutuhkan untuk proses saja.

1.2.3) Membuat Website Lebih Cepat dan Responsif

Dengan adanya AJAX, hanya data yang diperlukan saja yang akan direquest ke server. Maka, proses di server bisa jadi lebih cepat dan data bisa langsung dikirim kembali ke browser. Imbas yang dirasakan pengunjung adalah loading website jadi lebih cepat. Jadi, pengunjung tak perlu menunggu lama ketika mengakses fitur di website seperti chat, komentar, dan lainnya.

1.2.4) Callbacks

AJAX digunakan untuk melakukan callbacks, melakukan quick round trip ke dan dari server untuk mengambil atau menyimpan data tanpa memposting seluruh halaman kembali ke server. Dengan tidak melakukan postback penuh dan mengirim semua data formulir ke server, pemanfaatan jaringan diminimalkan dan operasi lebih cepat terjadi.

Di situs dan lokasi dengan bandwidth terbatas, cara ini dapat sangat meningkatkan kinerja jaringan. Sebagian besar waktu, data yang dikirim ke dan dari server sangat minimal. Dengan menggunakan callback, server tidak perlu memproses semua elemen formulir.

Dengan hanya mengirim data yang diperlukan, ada pemrosesan terbatas di server sehingga tidak perlu memproses semua elemen formulir, memproses kondisi tampilan, mengirim gambar kembali ke klien, atau mengirim halaman penuh kembali ke klien.

1.2.5) Membuat Asynchronous Calls

AJAX memungkinkan untuk melakukan Asynchronous Calls atau panggilan asinkron ke server web sehingga peramban klien tidak perlu menunggu semua data. Dikarenakan postback halaman dihilangkan, aplikasi yang diaktifkan AJAX akan selalu lebih responsif, lebih cepat dan lebih ramah pengguna.

1.2.6) Meningkatkan Kecepatan

Tujuan utama AJAX adalah untuk meningkatkan kecepatan, kinerja, dan kegunaan aplikasi web. Sebuah contoh dari AJAX yakni fitur peringkat film di Netflix. Kita bisa memberi peringkat film dan peringkat untuk film yang ditayangkan di Netflix yang kemudian akan disimpan ke database tanpa menunggu halaman diperbarui atau dimuat ulang. Peringkat film ini disimpan ke database tanpa memposting seluruh halaman kembali ke server. (Eril, 2020)

AJAX bukan teknologi tunggal, juga bukan bahasa pemrograman. Sistem umumnya terdiri dari:

1.2.1) HTML / XHTML untuk bahasa utama dan CSS untuk presentasi.

1.2.2) Document Object Model (DOM) untuk data tampilan dinamis dan interaksinya.

1.2.3) XML untuk pertukaran data dan XSLT untuk manipulasinya. Banyak pengembang sudah mulai mengganti dengan JSON karena lebih dekat ke JavaScript.

1.2.4) Objek XMLHttpRequest untuk digunakan sebagai komunikasi yang tidak sinkron. (C.Ariata, 2019)

1.3 Praktikum

Kode program berikut adalah contoh penggunaan AJAX untuk membuat saran otomatis pada text input:

```

<html>
<head>
<script>
function showHint(str) {
    if (str.length == 0) {
        document.getElementById("txtHint").innerHTML =
        ""; return;
    } else {
        var xmlhttp = new XMLHttpRequest();
        xmlhttp.onreadystatechange = function() {
            if (this.readyState == 4 && this.status == 200) {
                document.getElementById("txtHint").innerHTML =
                this.responseText;
            }
        };
        xmlhttp.open("GET", "gethint.php?q=" + str,
        true); xmlhttp.send();
    }
}
</script>
</head>
<body>

<p><b>Mulai ketikkan sebuah nama pada field input
dibawah:</b></p> <form>
Nama depan: <input type="text"
onkeyup="showHint(this.value)"> </form>
<p>Saran: <span id="txtHint"></span></p>
</body>
</html>

```

Gambar 1.1 Query Buka Koneksi

Pertama, program memeriksa apakah field input kosong (`str.length == 0`). Jika kosong, bersihkan konten dari placeholder `txtHint` dan keluar dari fungsi. Namun, jika field input tidak kosong, maka akan melakukan hal-hal sebagai berikut:

- 1.3.1) Membuat sebuah objek `XMLHttpRequest`.
- 1.3.2) Membuat fungsi untuk dieksekusi ketika jawaban server adalah siap (`ready`).
- 1.3.3) Mengirim permintaan off ke file PHP (`gethint.php`) pada server.
- 1.3.4) Memberitahukan bahwa parameter `q` ditambah ke URL (`gethint.php?q="+str`).
- 1.3.5) Variabel `str` menyimpan konten field input.

Kode program diatas menggunakan file PHP dengan nama “`gethint.php`”, kode program dari file tersebut adalah sebagai berikut:

```
<?php
// Array with names
$a[ ] = "Anna";
$a[ ] = "Brittany";
$a[ ] = "Cinderella";
$a[ ] = "Diana";
$a[ ] = "Eva";
$a[ ] = "Fiona";
$a[ ] = "Gunda";
$a[ ] = "Hege";
$a[ ] = "Inga";
$a[ ] = "Johanna";
$a[ ] = "Kitty";
$a[ ] = "Linda";
$a[ ] = "Nina";
$a[ ] = "Ophelia";
$a[ ] = "Petunia";
$a[ ] = "Amanda";
$a[ ] = "Raquel";
$a[ ] = "Cindy";
$a[ ] = "Doris";
$a[ ] = "Eve";
$a[ ] = "Evita";
$a[ ] = "Sunniva";
$a[ ] = "Tove";
$a[ ] = "Unni";
$a[ ] = "Violet";
$a[ ] = "Liza";
$a[ ] = "Elizabeth";
$a[ ] = "Ellen";
```

```

$a[ ] = "Wenche";
$a[ ] = "Vicky";
// get the q parameter from
URL $q = $_REQUEST["q"];
$hint = "";
// lookup all hints from array if $q is different from
"" if ($q !== "") {
$q = strtolower($q);
$len=strlen($q);
foreach($a as $name) {
if (striestr($q, substr($name, 0, $len)))
{ if ($hint === "") {
$hint = $name;
} else {
$hint .= ", $name";
}
}
}
}
// Output "no suggestion" if no hint was found or output correct
values echo $hint === "" ? "no suggestion" : $hint;
?>

```

Gambar 1.2 Field Input getHint.php

AJAX bisa digunakan untuk komunikasi interaktif dengan sebuah database. Dibawah ini akan diberikan contoh kode program untuk mengakses database menggunakan AJAX, namun sebelum itu, perlu dibuat tabel “user” pada database “my_db” dan diisi dengan data sebagai berikut:

id	FirstName	LastName	Age	Hometown	Job
1	Peter	Griffin	41	Quahog	Brewery
2	Lois	Griffin	40	Newport	Piano Teacher
3	Joseph	Swanson	39	Quahog	Police Officer
4	Glenn	Quagmire	41	Quahog	Pilot

Berikut adalah contoh program tersebut:

```
<html>
<head>
<script>
function showUser(str) {
if (str == "") {
document.getElementById("txtHint").innerHTML =
""; return;
} else {
if (window.XMLHttpRequest) {
// code for IE7+, Firefox, Chrome, Opera,
Safari xmlhttp = new XMLHttpRequest();
} else {
// code for IE6, IE5
xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange = function() {
if (this.readyState == 4 && this.status == 200) {
document.getElementById("txtHint").innerHTML = this.responseText; }
```



```

};
xmlhttp.open("GET","getuser.php?q="+str,true);
xmlhttp.send();
}
} <
/script>
</head>
<body>
<form>
<select name="users" onchange="showUser(this.value)">
<option value="">Select a person:</option>
<option value="1">Peter Griffin</option> <option
value="2">Lois Griffin</option>
<option value="3">Joseph Swanson</option>
<option value="4">Glenn Quagmire</option>
</select>
</form>
<br>
<div id="txtHint"><b>Person info will be listed here...</b></div>
</body>
</html>

```

Gambar 1.3 Program User Terpilih (getUser)

Penjelasan kode:

Pertama, memeriksa apakah ada data orang yang terpilih. Jika tidak ada maka (str== ""), bersihkan kontent dari txtHint dan keluar dari fungsi. Jika ada data orang terpilih, maka akan melakukan hal berikut:

- 1.3.1) Membuat objek XMLHttpRequest.
- 1.3.2) Membuat fungsi untuk dieksekusi ketika jawaban server adalah siap (ready).
- 1.3.3) Kirim permintaan off untuk sebuah file pada server.

1.3.4) Memberitahukan bahwa sebuah parameter (q) ditambahkan ke URL (dengan konten dropdown list).

Contoh program diatas menggunakan file “getuser.php”. File tersebut berisi kode program untuk melakukan query pada database MySQL, Kode program dari file tersebut adalah sebagai berikut:

```
<!DOCTYPE html>
<html>
<head>
<style>
table {
width: 100%;
border-collapse: collapse;
}
table, td, th {
border: 1px solid black;
padding: 5px;
}
th {text-align: left;}
</style>
</head>
<body>
<?php
$q = intval($_GET['q']);
$con = mysqli_connect('localhost','peter','abc123','my_db'); if
(!$con){
die('Could not connect: ' . mysqli_error($con));
}
mysqli_select_db($con,"ajax_demo");
$sql="SELECT * FROM user WHERE id =
'".$q."'"; $result = mysqli_query($con,$sql);
```

```

echo "<table>
<tr>
<th>Firstname</th>
<th>Lastname</th>
<th>Age</th>
<th>Hometown</th>
<th>Job</th>
</tr>";
while($row = mysqli_fetch_array($result)) {
echo "<tr>";
echo "<td>" . $row['FirstName'] . "</td>";
echo "<td>" . $row['LastName'] . "</td>";
echo "<td>" . $row['Age'] . "</td>";
echo "<td>" . $row['Hometown'] . "</td>";
echo "<td>" . $row['Job'] . "</td>";
echo "</tr>";
}
echo "</table>";
mysqli_close($con);
?>
</body>
</html>

```

Gambar 1.4 File getUser.php

Penjelasan dari kode program:

Ketika query dikirimkan dari JavaScript ke file PHP, hal-hal berikut terjadi:

- 1.3.1) PHP membuka koneksi ke server MySQL.
- 1.3.2) Menemukan data orang.
- 1.3.3) Tabel HTML dibuat, menambahkan data ke tabel tersebut, dan mengirimkannya kembali ke placeholder “txtHint”.

AJAX bisa digunakan untuk komunikasi interaktif dengan file XML. Berikut adalah contoh program AJAX yang digunakan untuk menampilkan data XML:

```
<html>
<head>
<script>
function showCD(str) {
if (str=="") {
document.getElementById("txtHint").innerHTML="";
return;
}
if (window.XMLHttpRequest) {
// code for IE7+, Firefox, Chrome, Opera,
Safari xmlhttp=new XMLHttpRequest();
} else { // code for IE6, IE5
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function() {
if (this.readyState==4 && this.status==200) {
document.getElementById("txtHint").innerHTML=this.responseText;
}
}
xmlhttp.open("GET","getcd.php?q="+str,true);
xmlhttp.send();
} <
</script>
</head>
<body>

<form>
Select a CD:
```

```

<select name="cds" onchange="showCD(this.value)">
<option value="">Select a CD:</option>
<option value="Bob Dylan">Bob Dylan</option>
<option value="Bee Gees">Bee Gees</option>
<option value="Cat Stevens">Cat Stevens</option>
</select>
</form>
<div id="txtHint"><b>CD info will be listed here...</b></div>
</body>
</html>

```

Gambar 1.5 showCD

Penjelasan kode program:

Fungsi showCD() melakukan hal-hal berikut:

- 1.3.1) Memeriksa apakah ada CD yang terpilih.
- 1.3.2) Membuat sebuah objek XMLHttpRequest.
- 1.3.3) Membuat fungsi yang akan dieksekusi ketika jawaban server adalah siap (ready).
- 1.3.4) Mengirimkan permintaan off untuk sebuah file pada server.
- 1.3.5) Memberitahukan bahwa sebuah parameter (q) ditambahkan ke URL (dengan konten dari list dropdown).

Pada kode program tersebut, terdapat file “getcd.php”. Kode program dari file tersebut memuat dokumen XML “cd_catalog.xml”, menjalankan query terhadap file XML dan mengembalikan hasilnya. Berikut adalah kode program dari file “getcd.php”:

```

<?php
$q=$_GET["q"];
$xmlDoc = new DOMDocument();
$xmlDoc->load("cd_catalog.xml");

$x=$xmlDoc->getElementsByTagName('ARTIST');

```

```

for ($i=0; $i<=$x->length-1; $i++) {
//Hanya memproses element node
if ($x->item($i)->nodeType==1){
if ($x->item($i)->childNodes->item(0)->nodeValue == $q)
{ $y=($x->item($i)->parentNode);
}
}
}
$cd=($y->childNodes);
for ($i=0;$i<$cd->length;$i++) {
//Hanya memproses element node
if ($cd->item($i)->nodeType==1){
echo("<b>" . $cd->item($i)->nodeName . ":</b> ");
echo($cd->item($i)->childNodes->item(0)-
>nodeValue); echo("<br>");
}
}
?>

```

Gambar 1.6 getCD

Ketika query CD dikirimkan dari JavaScript ke halaman PHP, hal-hal berikut terjadi:

- 1.3.1) PHP membuat sebuah objek XML DOM.
- 1.3.2) Menemukan semua element yang sesuai dengan nama yang dikirimkan dari JavaScript.
- 1.3.3) Mengeluarkan output berupa informasi album (mengirimkan ke placeholder “txtHint”)

AJAX bisa digunakan untuk membuat pencarian keyword yang lebih interaktif dan user-friendly. Berikut adalah contoh program untuk melakukan pencarian melalui query pada sebuah file XML “links.xml”. Kode program dari contoh program pencarian adalah sebagai berikut:

```
<html>
<head>
<script>
function showResult(str) {
if (str.length==0) {
document.getElementById("livesearch").innerHTML="";
document.getElementById("livesearch").style.border="0px";
return;
}
if (window.XMLHttpRequest) {
// code for IE7+, Firefox, Chrome, Opera,
Safari xmlhttp=new XMLHttpRequest();
} else { // code for IE6, IE5
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function() {
if (this.readyState==4 && this.status==200) {
document.getElementById("livesearch").innerHTML=this.responseText;
document.getElementById("livesearch").style.border="1px solid
#A5ACB2"; }
}
xmlhttp.open("GET","livesearch.php?q="+str,true);
xmlhttp.send();
} <
</script>
</head>
<body>

<form>
<input type="text" size="30" onkeyup="showResult(this.value)">
<div id="livesearch"></div>
```

```
</form>
</body>
</html>
```

Gambar 1.7 links.xml

Penjelasan kode program:

Jika field input kosong (str.length==0), fungsi akan membersihkan konten dari placeholder livesearch dan keluar dari fungsi. Jika field input tidak kosong, fungsi showResult() melakukan hal-hal berikut:

- 1.3.1) Membuat objek XMLHttpRequest.
- 1.3.2) Membuat fungsi yang akan dieksekusi ketika jawaban server adalah siap (ready).
- 1.3.3) Mengirimkan permintaan off untuk sebuah file pada server.
- 1.3.4) Memberitahukan bahwa sebuah parameter (q) ditambahkan ke URL (dengan konten field input).

Contoh kode program diatas menggunakan file “livesearch.php”. Kode program dari file tersebut adalah sebagai berikut:

```
<?php
$xmlDoc=new DOMDocument();
$xmlDoc->load("links.xml");
$x=$xmlDoc->getElementsByTagName('link');
//Mengambil parameter q dari URL
$q=$_GET["q"];
//lookup semua link dari file xml jika panjang dari
q>0 if (strlen($q)>0) {
$hint="";
for($i=0; $i<($x->length); $i++){
$y=$x->item($i)->getElementsByTagName('title');
$z=$x->item($i)->getElementsByTagName('url');
if ($y->item(0)->nodeType==1){
//menemukan sebuah link yang sesuai dengan text yang
dicari
```



```

if (strpos($y->item(0)->childNodes->item(0)-
>nodeValue,$q)){ if ($hint==""){
$hint="<a href=" .
$z->item(0)->childNodes->item(0)->nodeValue .
"" target='_blank'>" .
$y->item(0)->childNodes->item(0)->nodeValue . "</a>";
}else{
$hint=$hint . "<br /><a href=" .
$z->item(0)->childNodes->item(0)->nodeValue .
"" target='_blank'>" .
$y->item(0)->childNodes->item(0)->nodeValue .
"</a>"; }
}
}
}
}

// Mengatur output menjadi "no suggestion" jika tidak ada petunjuk
yang ditemukan atau nilai yang sesuai
if ($hint==""){
$response="no suggestion";
}else{
$response=$hint;
}

//output the response
echo $response;

?>

```

Gambar 1.8 livesearch.php

Penjelasan dari kode program:

Jika ada text yang dikirimkan dari JavaScript ($\text{strlen}(\$q) > 0$), hal-hal berikut akan dilakukan:

1.3.1) Memuat file XML menjadi objek XML DOM baru.

- 1.3.2) Perulangan melalui semua element untuk menemukan kata yang sesuai dengan text yang dikirimkan dari JavaScript.
- 1.3.3) Menugaskan URL yang benar dan judul dalam variabel \$response. Jika lebih dari satu yang sesuai, semua yang sesuai akan ditambahkan ke variabel tersebut.
- 1.3.4) Jika tidak ada yang sesuai, variabel \$response akan dimasukkan nilai text “no suggestion”.

AJAX juga baik digunakan untuk menampilkan secara langsung hasil polling. Berikut adalah contoh kode program untuk melakukan polling:

```
<html>
<head>
<script>
function getVote(int) {
if (window.XMLHttpRequest) {
// code for IE7+, Firefox, Chrome, Opera,
Safari xmlhttp=new XMLHttpRequest();
} else { // code for IE6, IE5
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function() {
if (this.readyState==4 && this.status==200) {
document.getElementById("poll").innerHTML=this.responseText;
}
}
xmlhttp.open("GET","poll_vote.php?vote="+int,true);
xmlhttp.send();
} <
</script>
</head>
<body>
```

```

<div id="poll">
<h3>Apakah kamu menyukai AJAX atau PHP?
</h3> <form>
Yes:
<input type="radio" name="vote" value="0"
onclick="getVote(this.value)"> <br>No:
<input type="radio" name="vote" value="1" onclick="getVote(this.value)">
</form>
</div>
</body>
</html>

```

Gambar 1.9 getVote

Fungsi getVote() melakukan hal-hal sebagai berikut:

- 1.3.1) Membuat sebuah objek XMLHttpRequest
- 1.3.2) Membuat fungsi untuk dieksekusi ketika jawaban server adalah siap (ready).
- 1.3.3) Mengirimkan permintaan off untuk sebuah file pada server.
- 1.3.4) Memberitahukan bahwa parameter (vote) ditambahkan ke URL (dengan nilai opsi yes atau no).

Kode program diatas menggunakan file “poll_vote.php”. Kode program dari file tersebut adalah sebagai berikut:

```

<?php
$vote = $_REQUEST['vote'];
//mendapatkan kontent dari textfile
$filename = "poll_result.txt";
$content = file($filename);
//meletakkan kontent dalam array
$array = explode("||", $content[0]);
$yes = $array[0];
$no = $array[1];

```

```
if ($vote == 0){
$yes = $yes + 1;
}
if ($vote == 1){
$no = $no + 1;
}

//menyisipkan vote ke file txt
$insertvote = $yes."||".$no;
$fp = fopen($filename,"w");
fputs($fp,$insertvote);
fclose($fp);
?>
<h2>Result:</h2>
<table>
<tr>
<td>Yes:</td>
<td>

' height='20'>
<?php echo(100*round($yes/($no+$yes),2)); ?>%
</td>
</tr>
<tr>
<td>No:</td>
<td>
' height='20'>
<?php echo (100*round($no/($no+$yes),2)); ?>%
</td>
```

```
</tr>  
</table>
```

Gambar 1.10 poll_vote.php

Nilai dikirim dari JavaScript dan hal-hal berikut akan terjadi:

- 1.3.1) Mengambil konten dari file “poll_result.txt”.
- 1.3.2) Meletakkan konten ke dalam variabel dan menambahkan satu ke variabel yang terpilih.
- 1.3.3) Menuliskan hasil ke file “poll_result.txt”
- 1.3.4) Mengeluarkan representasi grafis dari hasil polling.

File text (poll_result.txt) adalah tempatn menyimpan data polling. Data tersebut disimpan dalam bentuk berikut:

0||0

Bilangan pertama merepresentasikan voting “Yes” sedangkan bilangan kedua merepresentasikan voting “No”. Jangan lupa untuk mengizinkan web browser merubah text file.(Dosen Teknik Informatika, 2021)

BAB II

PEMBAHASAN

2.1 Langkah Kerja

- 2.1.1) Buatlah program web untuk menyimpan data survei yang disimpan didalam database. Carilah library javascript diinternet untuk menampilkan grafik, kemudian rangkum data-data hasil survei kedalam bentuk grafik.
- 2.1.2) Misalkan anda seorang programmer yang diminta untuk membuat program web untuk melakukan pemungutan suara pada PEMILU. Jumlah calon yang akan dipilih adalah lima orang. Buatlah program web untuk pemilihan tersebut, dimana setiap pemilih hanya bisa memilih satu kali.

2.2 Pembahasan

2.2.1 Survei Pengunjung Perpustakaan

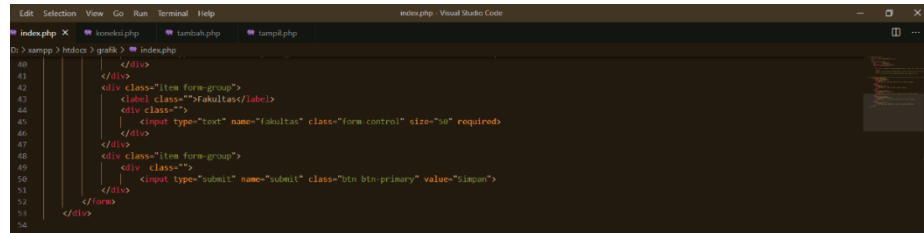
```

Edit Selection View Go Run Terminal Help
index.php - Visual Studio Code

index.php x koneksi.php tambah.php tampil.php
D:\> xamp> htdocs > grafik > index.php
1 <?php include("koneksi.php"); ?>
2
3 <center><font size="6">Tambah Data</font></center>
4 <br>
5 <?php
6 if(isset($_POST['submit'])){
7     $nama      = $_POST['nama'];
8     $nim       = $_POST['nim'];
9     $jenis_kelamin = $_POST['jenis_kelamin'];
10    $fakultas  = $_POST['fakultas'];
11
12
13    $sql = mysql_query($koneksi, "INSERT INTO mahasiswa VALUES ('','$nama','$nim','$jenis_kelamin','$fakultas')") or die(mysql_error($koneksi));
14
15    if($sql){
16        echo "<script>alert('Berhasil menambahkan data.');" document.location='tampil.php';</script>";
17    }else{
18        echo "<div class='alert alert-warning'>Gagal melakukan proses tambah data.</div>";
19    }
20
21 ?>
22
23 <form action="" method="post">
24     <div class="item form-group">
25         <label class=">>nama</label>
26         <div class="
27             <input type="text" name="nama" class="form-control" size="50" required>
28         </div>
29     </div>
30     <div class="
31         <label class=">>nim</label>
32         <div class="
33             <input type="text" name="nim" class="" size="50" required>
34         </div>
35     </div>
36     <div class="item form-group">
37         <label class=">>jenis kelamin</label>
38         <div class="
39             <input type="text" name="jenis_kelamin" class="form-control" size="50" required>

```

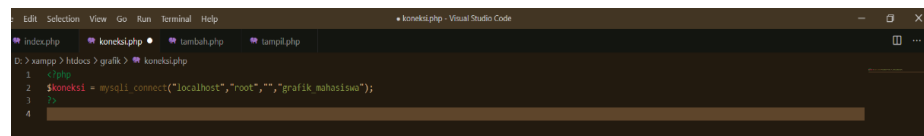
Gambar 2.1 index.php



Gambar 2.2 index.php

Blok kode di atas adalah blok kode dari file index.php. Fungsi dari file index ini ialah sebagai lokasi untuk menampilkan menu tambah data yang ada pada web sederhana. Selain itu, index juga berfungsi untuk memberikan tempat input data-data milik pengunjung perpustakaan yang nantinya akan ditampilkan. Berikut penjelasan kode-kodenya.

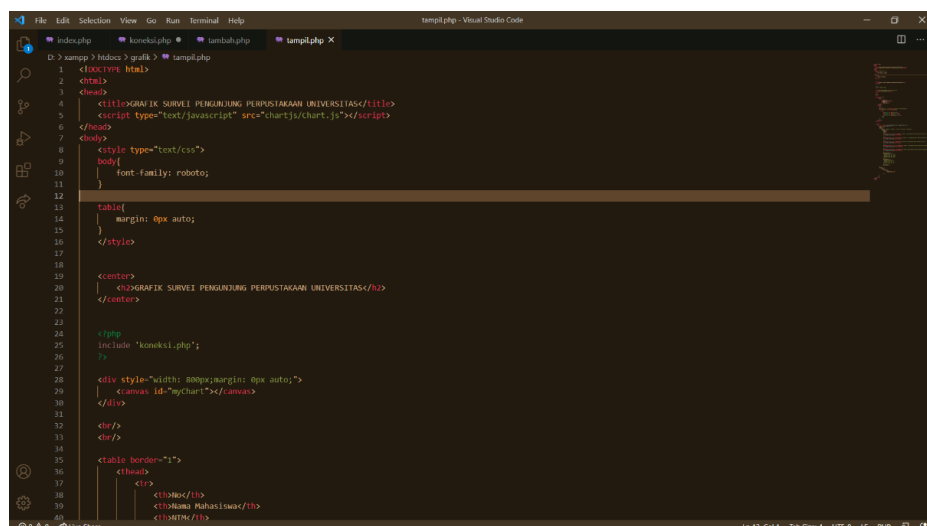
- 1) Fungsi PHP include() merupakan fungsi yang digunakan untuk menyertakan file php lain ke dalam suatu program PHP. Hal sangat membantu proses pemrograman karena tidak perlu menulis program PHP secara berulang-ulang, cukup dalam satu file saja (Indra, 2014).
- 2) Fungsi isset pada PHP biasanya digunakan untuk mengecek form dari HTML. Sedangkan variabel \$_POST digunakan jika pengiriman data dari form menggunakan method POST. Tujuan dari fungsi isset pada php ini untuk menghindari error undefined variable (variabel tidak didefinisi) (Lukman, 2019).
- 3) Form adalah penunjuk bahwa data berbentuk seperti formulir saat akan diisi. Form action merupakan form handling yang nantinya data yang dapat dihandling melalui action. Action method ini berupa \$_POST (Hadi, 2015).
- 4) \$query digunakan untuk menampung perintah yang diberikan kepada data yang ada di database berdasarkan dengan kondisi yang telah ditentukan.
- 5) Type digunakan untuk penunjuk tipe data yang digunakan saat akan menginput data ke dalam database.



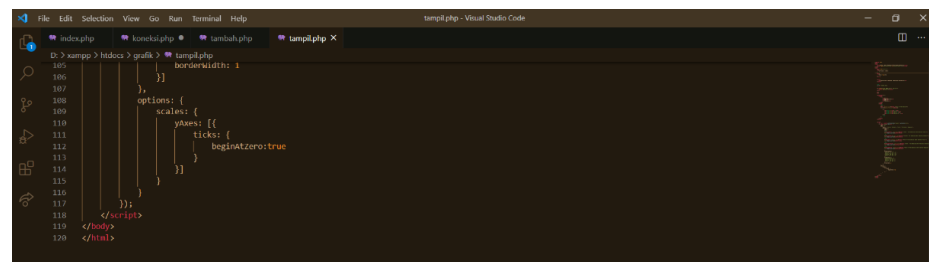
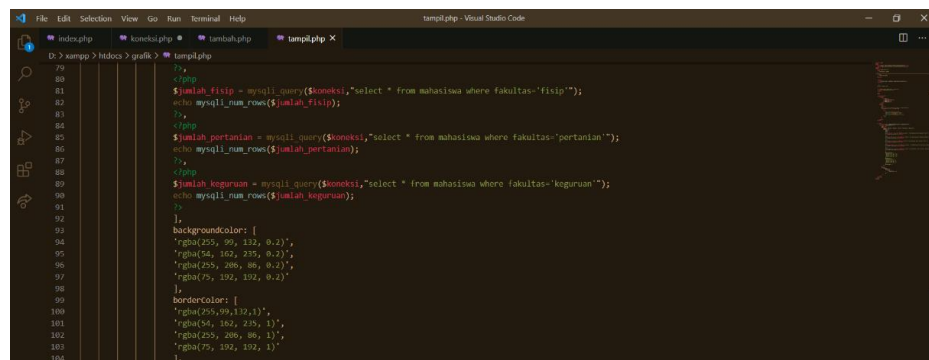
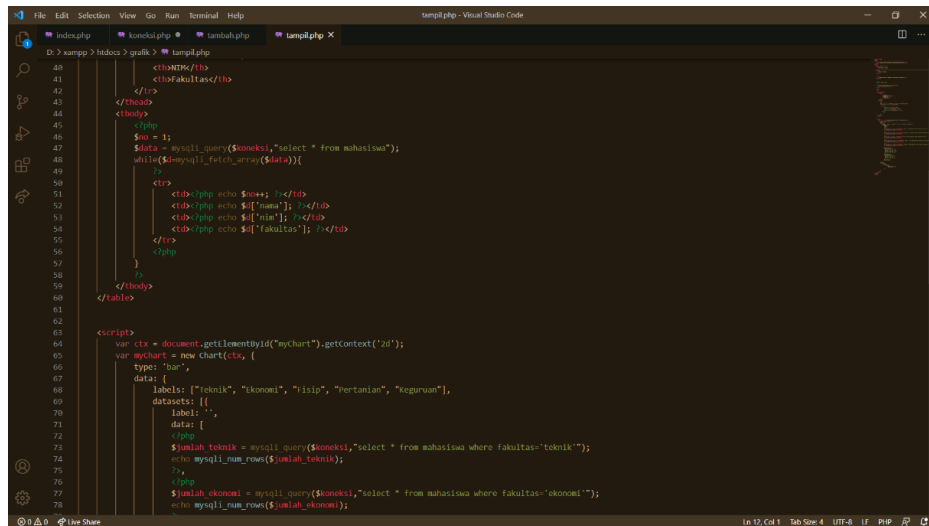
Gambar 2.3 koneksi.php

Blok kode di atas adalah blok kode dari file koneksi.php. File php ini berfungsi untuk menghubungkan antara program dari file php dengan database sql yang digunakan. Dalam penggunaanya, terdapat kode-kode seperti berikut.

- 1) <?php : Sintaks ini adalah kode wajib untuk membuka program PHP.
- 2) \$koneksi berfungsi sebagai penghubung antara php koneksi dengan php php lainnya yang memiliki array yang sama dengan yang ada di dalam parameter \$koneksi.
- 3) Fungsi mysqli_connect() merupakan fungsi yang digunakan untuk membuka koneksi ke server MySQL dan memilih database yang akan digunakan.
- 4) ?> : Sintaks ini adalah kode untuk mengakhiri PHP dan wajib digunakan saat digabung dengan bahasa pemrograman lain seperti HTML (Awwaabiin, 2020).



Gambar 2.4 tampil.php



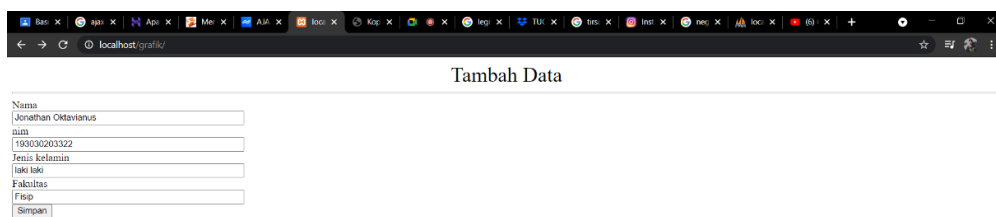
Blok kode di atas adalah blok kode dari file tampil.php. File php ini berfungsi untuk menampilkan grafik yang nantinya menunjukkan banyak pengunjung perpustakaan yang berasal dari berbagai macam fakultas. Berikut ini adalah penjelasan kode-kodenya.

- 1) `<!DOCTYPE html>`; adalah suatu deklarasi yang digunakan untuk mengidentifikasi jenis dokumen HTML yang digunakan sehingga

browser dapat menentukan bagaimana memperlakukan kode tersebut.

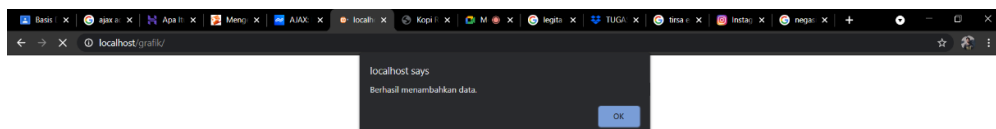
- 2) `<html>`; adalah tag utama yang digunakan untuk memulai dokumen HTML.
- 3) `<head>`; adalah tag utama yang digunakan untuk membuat bagian kepala dokumen.
- 4) `<title>`; adalah tag utama yang digunakan untuk mengatur judul web.
- 5) `<body>`; adalah tag utama yang digunakan untuk membuat bagian tubuh dokumen (Muhardian, 2020).
- 6) Atribut `border` digunakan untuk mengatur ketebalan dari garis tepi (`border`) dari tabel. Jika atribut ini tidak ditulis, maka web browser akan menampilkan tabel tanpa garis tepi. Nilai dari atribut ini berupa angka yang diukur dalam satuan pixel. Jika anda memberikan nilai `border="1"`, maka web browser akan menampilkan garis tepi sebesar 1 pixel pada sisi atas, bawah, kiri dan kanan table (Andre, 2013).
- 7) Elemen `<tr>` digunakan untuk mendefinisikan pembuatan baris pada tabel
- 8) Elemen `<td>` digunakan untuk membuat kolom atau sel di setiap baris pada table.

Sehingga, didapat output atau hasil web sederhana yang menampilkan hasil dari kode-kode file php di atas seperti yang ditunjukkan pada gambar berikut.

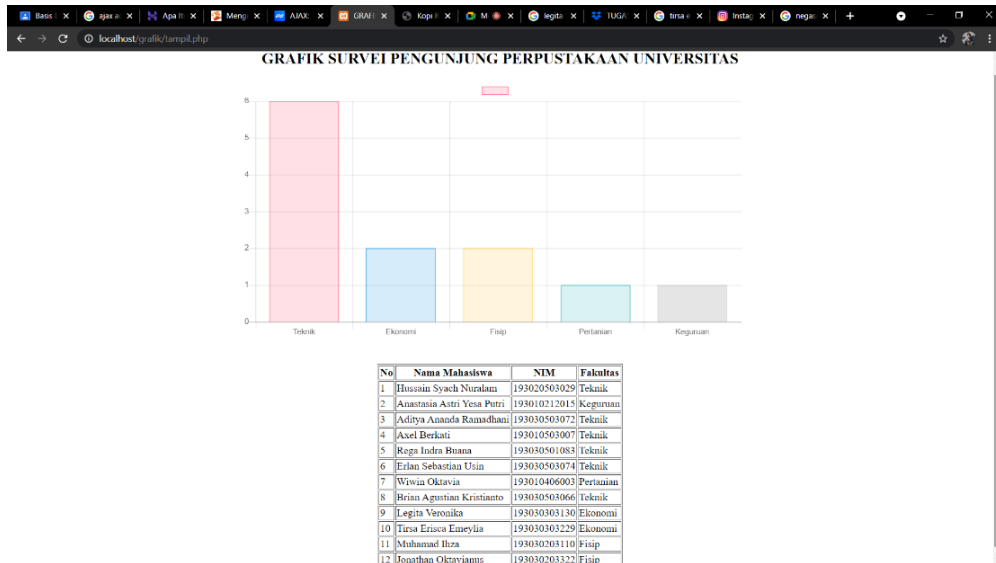


The screenshot shows a web browser window with the address bar displaying 'localhost/grafik/'. The page title is 'Tambah Data'. Below the title is a form with the following fields: 'Nama' (filled with 'Jonathan Oktavianus'), 'nim' (filled with '19930203322'), 'Jenis kelamin' (filled with 'laki laki'), 'Fakultas' (filled with 'Fisip'), and a 'Simpan' button.

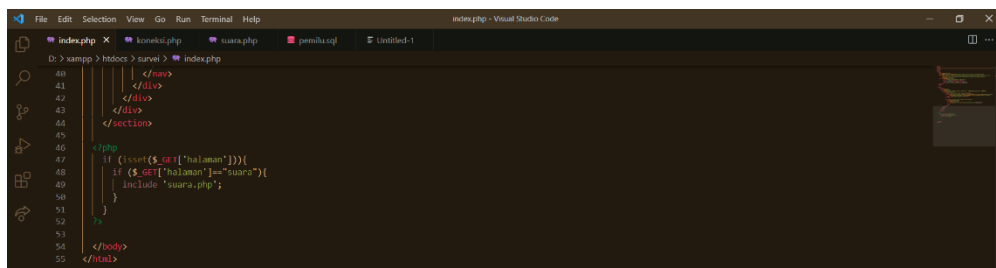
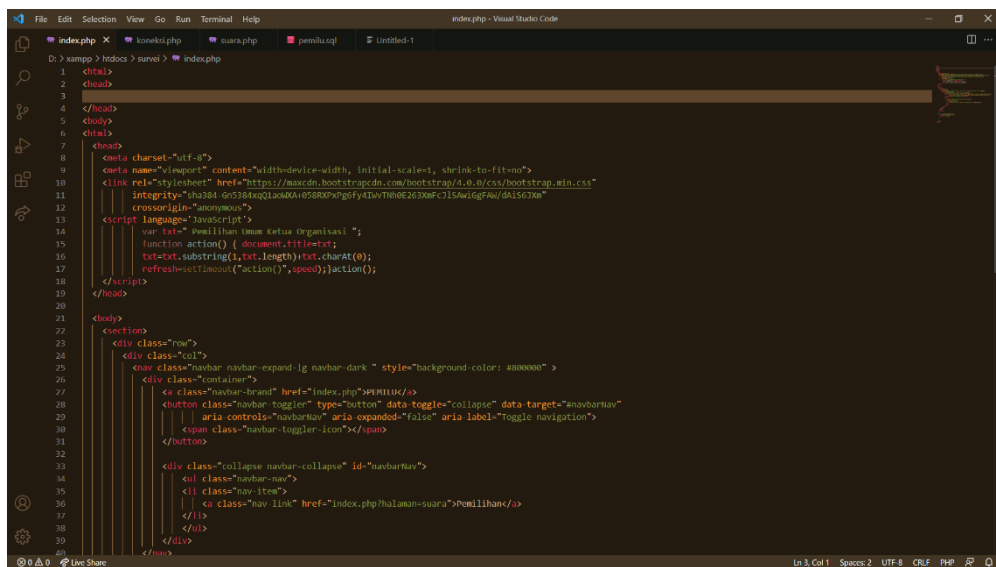
Gambar 2.8 Laman Tambah Data



Gambar 2.9 Data Berhasil Ditambahkan

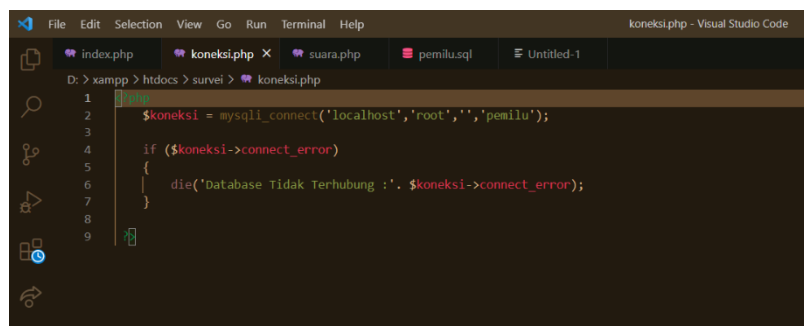


2.2.2 Web Sederhana Voting Pemilu



Blok kode yang ada di atas adalah blok kode dari file index.php. Fungsi dari file index ini ialah sebagai lokasi untuk menampilkan menu dari program web voting. Dalam index ini terdapat nav yang merupakan blok kode yang menghasilkan tombol navigasi. Berikut ini adalah penjelasan kode-kodenya.

- 1) <html>; adalah tag utama yang digunakan untuk memulai dokumen HTML.
- 2) <head>; adalah tag utama yang digunakan untuk membuat bagian kepala dokumen.
- 3) <body>; adalah tag utama yang digunakan untuk membuat bagian tubuh dokumen (Muhardian, 2020).
- 4) Fungsi isset pada PHP biasanya digunakan untuk mengecek form dari HTML. Sedangkan variabel \$_GET digunakan jika menerima data dari form menggunakan method GET. Tujuan dari fungsi isset pada php ini untuk menghindari error undefined variable (variabel tidak didefinisi) (Lukman, 2019).
- 5) <?php : Sintaks ini adalah kode wajib untuk membuka program PHP.
- 6) ?> : Sintaks ini adalah kode untuk mengakhiri PHP dan wajib digunakan saat digabung dengan bahasa pemrograman lain seperti HTML (Awwaabiin, 2020).
- 7) <div> tag element dapat dijadikan "wadah" untuk menempatkan beberapa elemen menjadi satu bagian, yang penggunaannya ditujukan untuk mempermudah aplikasi style dengan CSS dan perlakuan khusus lainnya yang kemudian dapat diberi atribut class, id, title dan lain sebagainya.



Gambar 2.13 koneksi.php

Blok kode di atas adalah blok kode dari file koneksi.php. File php ini berfungsi untuk menghubungkan antara program dari file php dengan database sql yang digunakan. Dalam penggunaannya, terdapat kode-kode seperti berikut.

- 1) <?php : Sintaks ini adalah kode wajib untuk membuka program PHP.
- 2) \$koneksi berfungsi sebagai penghubung antara php koneksi dengan php php lainnya yang memiliki array yang sama dengan yang ada di dalam parameter \$koneksi.
- 3) Fungsi mysqli_connect() merupakan fungsi yang digunakan untuk membuka koneksi ke server MySQL dan memilih database yang akan digunakan.
- 4) ?> : Sintaks ini adalah kode untuk mengakhiri PHP dan wajib digunakan saat digabung dengan bahasa pemrograman lain seperti HTML (Awwaabiin, 2020).

```

1 <?php
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
6 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
7 integrity="sha384-G554XG1dUDXqJM599UMFH8prB60J44ZEN6ObfcS0a45DI48j740ZF63B70AFK2f35"
8 crossorigin="anonymous">
9 </head>
10 <body>
11 <div>
12 <h1>PEMERINTAH URUM KETUA ORGANISASI</h1>
13 </div>
14 <form method="post">
15 <select name="pilih">
16 <option value="1">1. Masine</option>
17 <option value="2">2. Elnata</option>
18 <option value="3">3. Elnata</option>
19 <option value="4">4. Aditya</option>
20 <option value="5">5. Anel</option>
21 </select>
22 <label id="pilih">Label</label>
23 <input type="text" name="id_pilih"/>
24 <button name="kirin" type="submit">kirin</button>
25 </form>
26 <?php
27 require './koneksi.php';
28 if (isset($_POST['kirin'])) {
29 // menangkap data yang di kirin dari form
30 $pilih = $_POST['pilih'];
31 $id_pilih = $_POST['id_pilih'];
32
33 $data = mysqli_query($koneksi,"select * from suara where id_pilih=$id_pilih");
34
35 // menghitung jumlah data yang ditonkan
36 $cek = mysqli_num_rows($data);
37 echo $cek;
38 if ($cek <= 0){
39
40 $data = mysqli_query($koneksi,"INSERT INTO 'suara' ('id_suara','id_pilih','pilih','waktu') VALUES ('','$id_pilih','$pilih',CURRENT_TIMESTAMP)");

```

Gambar 2.14 suara.php

```

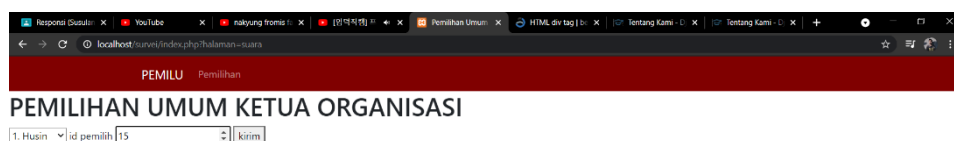
41
42 if ($data) {
43 <script language="javascript">
44 alert("Data berhasil ditambah");
45 </script>
46 <?php
47
48 }else if ($cek >= 0){
49 <script language="javascript">
50 alert("Masal id sudah digunakan");
51 </script>
52 <?php
53
54 }
55
56 </body>
57 </html>

```

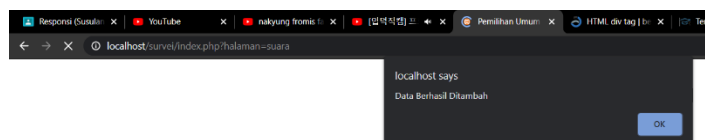
Gambar 2.15 suara.php

Blok kode di atas adalah blok kode dari file suara.php. File ini berfungsi untuk menampilkan menu pemilihan suara (voting) ketua organisasi yang mana data yang dipilih dan diisi akan disimpan ke dalam database. Hal ini dikarenakan adanya fungsi require yang berarti bahwa file suara.php ini dapat berjalan apabila terdapat file koneksi.php di dalam satu folder yang sama dengan suara.php. 1) PHP secara berulang-ulang, cukup dalam satu file saja (Indra, 2014). Fungsi isset pada PHP biasanya digunakan untuk mengecek form dari HTML. Sedangkan variabel \$_POST digunakan jika pengiriman data dari form menggunakan method POST. Tujuan dari fungsi isset pada php ini untuk menghindari error undefined variable (variabel tidak didefinisi) (Lukman, 2019). Alert digunakan untuk menunjukkan peringatan pada web apakah id yang dimasukan berhasil dimasukan atau tidak (dikarenakan sudah terpakai).

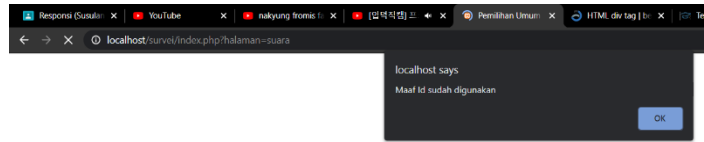
Berikut ini adalah output dari web voting pemilihan umum ketua organisasi.



Gambar 2.16 Web Voting



Gambar 2.17 Web Voting



Gambar 2.18 Web Voting

BAB III

KESIMPULAN

AJAX atau Asynchronous JavaScript and XML adalah teknik yang digunakan untuk membuat website yang dinamis. Artinya website mampu mengupdate dan menampilkan data baru dari server tanpa perlu melakukan reload. Sesuai namanya, AJAX terdiri dari JavaScript dan XML yang bekerja bersama. JavaScript adalah bahasa pemrograman untuk mengelola konten website yang dinamis. Sementara XML (eXtensible Markup Language) digunakan untuk memuat dan membawa data dari server ke browser.

AJAX JavaScript dan XML ini bekerja secara asynchronous untuk berkomunikasi dengan server. Proses pertukaran informasi ini dilakukan di background. Artinya, saat AJAX JavaScript dan XML bekerja, halaman dapat tetap diakses oleh pengunjung website. AJAX memungkinkan untuk melakukan Asynchronous Calls atau panggilan asinkron ke server web sehingga peramban klien tidak perlu menunggu semua data.

DAFTAR PUSTAKA

- Andre. (2013). *Atribut border, cellpadding dan cellspacing dalam Tabel*.
[Https://Www.Duniaikom.Com/](https://Www.Duniaikom.Com/). <https://www.duniaikom.com/belajar-html-mengenal-attribut-border-cellpadding-dan-cellspacing-dalam-tabel/>
- Aprilia, P. (2021). *AJAX-Mengenal Pengertian, Fungsi, dan Cara Kerjanya*.
Niagahoster.Co.Id. <https://www.niagahoster.co.id/blog/ajax-javascript/>
- Awwaabiin, S. (2020). *Apa itu PHP? Pengertian, Fungsi dan Sintaks Dasarnya!*
Niagahoster.Co.Id. <https://www.niagahoster.co.id/blog/pengertian-php/>
- C, A. (2019). *Apa Itu AJAX dan Bagaimana Cara Kerjanya?* Hostinger.
<https://www.hostinger.co.id/tutorial/apa-itu-ajax/>
- Dosen Teknik Informatika. (2021). *MODUL PRAKTIKUM PEMROGRAMAN WEB I. 1*, 60.
- Eril. (2020). *Mengenal Lebih Lengkap Tentang AJAX*. Qwords.Com.
<https://qwords.com/blog/ajax-adalah/>
- Hadi, D. A. (2015). *Penanganan Form Pada PHP*.
<Https://Www.Malasngoding.Com/>. <https://www.malasngoding.com/belajar-php-penanganan-form-pada-php/>
- Indra, R. (2014). *Mengenal Fungsi Require Dan Include Dalam PHP*.
<https://www.dumetschool.com/blog/Mengenal-Fungsi-Required-Dan-Include-Dalam-PHP#>
- Lukman. (2019). *Penjelasan Fungsi Isset Pada PHP*. <Https://Carakode.Com/>.
<https://carakode.com/penjelasan-fungsi-isset-pada-php/>
- Muhardian, A. (2020). *Belajar HTML #02 Apa itu Tag, Elemen, dan Atribut dalam HTML*. Petanikode.Com. <https://www.petanikode.com/html-tag-elemen-attribut/>

LAMPIRAN

```
<html>
<head>
<script>
function showHint(str) {
    if (str.length == 0) {
        document.getElementById("txtHint").innerHTML =
            ""; return;
    } else {
        var xmlhttp = new XMLHttpRequest();
        xmlhttp.onreadystatechange = function() {
            if (this.readyState == 4 && this.status == 200) {
                document.getElementById("txtHint").innerHTML =
this.responseText;
            }
        };
        xmlhttp.open("GET", "gethint.php?q=" + str,
            true); xmlhttp.send();
    }
}
</script>
</head>
<body>

<p><b>Mulai ketikkan sebuah nama pada field input
dibawah:</b></p> <form>
Nama depan: <input type="text"
onkeyup="showHint(this.value)"> </form>
<p>Saran: <span id="txtHint"></span></p>
</body>
</html>
```

Gambar 1.1 Query Buka Koneksi

```
<?php
// Array with names
$a[ ] = "Anna";
$a[ ] = "Brittany";
$a[ ] = "Cinderella";
$a[ ] = "Diana";
$a[ ] = "Eva";
$a[ ] = "Fiona";
$a[ ] = "Gunda";
$a[ ] = "Hege";
$a[ ] = "Inga";
$a[ ] = "Johanna";
$a[ ] = "Kitty";
$a[ ] = "Linda";
$a[ ] = "Nina";
$a[ ] = "Ophelia";
$a[ ] = "Petunia";
$a[ ] = "Amanda";
$a[ ] = "Raquel";
$a[ ] = "Cindy";
$a[ ] = "Doris";
$a[ ] = "Eve";
$a[ ] = "Evita";
$a[ ] = "Sunniva";
$a[ ] = "Tove";
$a[ ] = "Unni";
$a[ ] = "Violet";
$a[ ] = "Liza";
$a[ ] = "Elizabeth";
$a[ ] = "Ellen";
```

```
$a[ ] = "Wenche";
$a[ ] = "Vicky";
// get the q parameter from
URL $q = $_REQUEST["q"];
$hint = "";
// lookup all hints from array if $q is different from
"" if ($q !== "") {
    $q = strtolower($q);
    $len=strlen($q);
    foreach($a as $name) {
        if (stristr($q, substr($name, 0, $len)))
        { if ($hint === "") {
            $hint = $name;
        } else {
            $hint .= ", $name";
        }
    }
}
// Output "no suggestion" if no hint was found or output correct
values echo $hint === "" ? "no suggestion" : $hint;
?>
```

Gambar 1.2 Field Input getHint.php

```
<html>
<head>
<script>
function showUser(str) {
if (str == "") {
document.getElementById("txtHint").innerHTML =
""; return;
} else {
if (window.XMLHttpRequest) {
// code for IE7+, Firefox, Chrome, Opera,
Safari xmlhttp = new XMLHttpRequest();
} else {
// code for IE6, IE5
xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange = function() {
if (this.readyState == 4 && this.status == 200) {
document.getElementById("txtHint").innerHTML = this.responseText; }
}
```

Gambar 1.3 Program User Terpilih (getUser)

```

};
xmlhttp.open("GET","getuser.php?q="+str,true);
xmlhttp.send();
}
} <
/script>
</head>
<body>
<form>
<select name="users" onchange="showUser(this.value)">
<option value="">Select a person:</option>
<option value="1">Peter Griffin</option> <option
value="2">Lois Griffin</option>
<option value="3">Joseph Swanson</option>
<option value="4">Glenn Quagmire</option>
</select>
</form>
<br>
<div id="txtHint"><b>Person info will be listed here...</b></div>
</body>
</html>

```

Gambar 1.3 Program User Terpilih (getUser)

```

<!DOCTYPE html>
<html>
<head>
<style>
table {
width: 100%;
border-collapse: collapse;
}
table, td, th {
border: 1px solid black;
padding: 5px;
}
th {text-align: left;}
</style>
</head>
<body>
<?php
$q = intval($_GET['q']);
$con = mysqli_connect('localhost','peter','abc123','my_db'); if
(!$con){
die('Could not connect: ' . mysqli_error($con));
}
mysqli_select_db($con,"ajax_demo");
$sql="SELECT * FROM user WHERE id =
".$_q.""; $result = mysqli_query($con,$sql);

```

Gambar 1.4 File getUser.php

```

echo "<table>
<tr>
<th>Firstname</th>
<th>Lastname</th>
<th>Age</th>
<th>Hometown</th>
<th>Job</th>
</tr>";
while($row = mysqli_fetch_array($result)) {
echo "<tr>";
echo "<td>" . $row['FirstName'] . "</td>";
echo "<td>" . $row['LastName'] . "</td>";
echo "<td>" . $row['Age'] . "</td>";
echo "<td>" . $row['Hometown'] . "</td>";
echo "<td>" . $row['Job'] . "</td>";
echo "</tr>";
}
echo "</table>";
mysqli_close($con);
?>
</body>
</html>

```

Gambar 1.4 File getUser.php


```

<html>
<head>
<script>
function showCD(str) {
if (str=="") {
document.getElementById("txtHint").innerHTML="";
return;
}
if (window.XMLHttpRequest) {
// code for IE7+, Firefox, Chrome, Opera,
Safari xmlhttp=new XMLHttpRequest();
} else { // code for IE6, IE5
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function() {
if (this.readyState==4 && this.status==200) {
document.getElementById("txtHint").innerHTML=this.responseText;
}
}
xmlhttp.open("GET","getcd.php?q="+str,true);
xmlhttp.send();
} <
/script>
</head>
<body>

<form>
Select a CD:

```

Gambar 1.5 showCD

```
<select name="cds" onchange="showCD(this.value)">
<option value="">Select a CD:</option>
<option value="Bob Dylan">Bob Dylan</option>
<option value="Bee Gees">Bee Gees</option>
<option value="Cat Stevens">Cat Stevens</option>
</select>
</form>
<div id="txtHint"><b>CD info will be listed here...</b></div>
</body>
</html>
```

Gambar 1.5 showCD

```

<?php
$q=$_GET["q"];
$xmlDoc = new DOMDocument();
$xmlDoc->load("cd_catalog.xml");

$x=$xmlDoc->getElementsByTagName('ARTIST');

for ($i=0; $i<=$x->length-1; $i++) {
//Hanya memproses element node
if ($x->item($i)->nodeType==1){
if ($x->item($i)->childNodes->item(0)->nodeValue == $q)
{ $y=($x->item($i)->parentNode);
}
}
}
$cd=($y->childNodes);
for ($i=0;$i<$cd->length;$i++) {
//Hanya memproses element node
if ($cd->item($i)->nodeType==1){
echo("<b>" . $cd->item($i)->nodeName . ":</b> ");
echo($cd->item($i)->childNodes->item(0)-
>nodeValue); echo("<br>");
}
}
?>

```

Gambar 1.6 getCD

```
<html>
<head>
<script>
function showResult(str) {
if (str.length==0) {
document.getElementById("livesearch").innerHTML="";
document.getElementById("livesearch").style.border="0px";
return;
}
if (window.XMLHttpRequest) {
// code for IE7+, Firefox, Chrome, Opera,
Safari xmlhttp=new XMLHttpRequest();
} else { // code for IE6, IE5
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function() {
if (this.readyState==4 && this.status==200) {
document.getElementById("livesearch").innerHTML=this.responseText;
document.getElementById("livesearch").style.border="1px solid
#A5ACB2"; }
}
xmlhttp.open("GET","livesearch.php?q="+str,true);
xmlhttp.send();
} <
</script>
</head>
<body>

<form>
<input type="text" size="30" onkeyup="showResult(this.value)">
<div id="livesearch"></div>
```

```
</form>
</body>
</html>
```

Gambar 1.7 links.xml

```
<?php
$xmlDoc=new DOMDocument();
$xmlDoc->load("links.xml");
$x=$xmlDoc->getElementsByTagName('link');
//Mengambil parameter q dari URL
$q=$_GET["q"];
//lookup semua link dari file xml jika panjang dari
q>0 if (strlen($q)>0) {
$hint="";
for($i=0; $i<($x->length); $i++){
$y=$x->item($i)->getElementsByTagName('title');
$z=$x->item($i)->getElementsByTagName('url');
if ($y->item(0)->nodeType==1){
//menemukan sebuah link yang sesuai dengan text yang
dicari
```

Gambar 1.8 livesearch.php

```

if (strpos($y->item(0)->childNodes->item(0)-
>nodeValue,$q)){ if ($hint==""){
$hint="<a href=" .
$z->item(0)->childNodes->item(0)->nodeValue .
"" target='_blank'>" .
$y->item(0)->childNodes->item(0)->nodeValue . "</a>";
}else{
$hint=$hint . "<br /><a href=" .
$z->item(0)->childNodes->item(0)->nodeValue .
"" target='_blank'>" .
$y->item(0)->childNodes->item(0)->nodeValue .
"</a>"; }
}
}
}
}

// Mengatur output menjadi "no suggestion" jika tidak ada petunjuk
yang ditemukan atau nilai yang sesuai
if ($hint==""){
$response="no suggestion";
}else{
$response=$hint;
}

//output the response
echo $response;

?>

```

Gambar 1.8 livesearch.php

```
<html>
<head>
<script>
function getVote(int) {
if (window.XMLHttpRequest) {
// code for IE7+, Firefox, Chrome, Opera,
Safari xmlhttp=new XMLHttpRequest();
} else { // code for IE6, IE5
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function() {
if (this.readyState==4 && this.status==200) {
document.getElementById("poll").innerHTML=this.responseText;
}
}
xmlhttp.open("GET","poll_vote.php?vote="+int,true);
xmlhttp.send();
} <
/script>
</head>
<body>
```

Gambar 1.9 getVote

```
<div id="poll">
<h3>Apakah kamu menyukai AJAX atau PHP?
</h3> <form>
Yes:
<input type="radio" name="vote" value="0"
onclick="getVote(this.value)"> <br>No:
<input type="radio" name="vote" value="1" onclick="getVote(this.value)">
</form>
</div>
</body>
</html>
```

Gambar 1.9 getVote

```
<?php
$vote = $_REQUEST['vote'];
//mendapatkan konten dari textfile
$filename = "poll_result.txt";
$content = file($filename);
//meletakkan konten dalam array
$array = explode("||", $content[0]);
$yes = $array[0];
$no = $array[1];
```

Gambar 1.10 poll_vote.php


```

if ($vote == 0){
$yes = $yes + 1;
}
if ($vote == 1){
$no = $no + 1;
}
//menyisipkan vote ke file txt
$insertvote = $yes."||".$no;
$fp = fopen($filename,"w");
fputs($fp,$insertvote);
fclose($fp);
?>
<h2>Result:</h2>
<table>
<tr>
<td>Yes:</td>
<td>

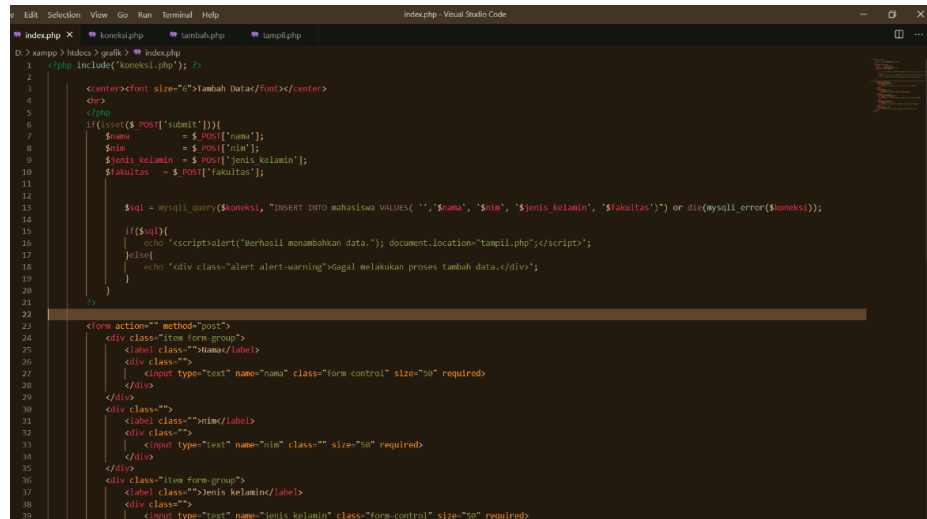
' height='20'>
<?php echo(100*round($yes/($no+$yes),2)); ?>%
</td>
</tr>
<tr>
<td>No:</td>
<td>
' height='20'>
<?php echo (100*round($no/($no+$yes),2)); ?>%
</td>

```

Gambar 1.10 poll_vote.php

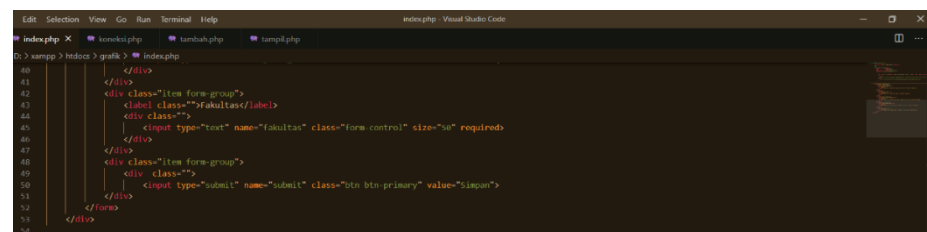
```
</tr>
</table>
```

Gambar 1.10 poll_vote.php

A screenshot of a Visual Studio Code editor window showing the code for index.php. The code includes a PHP script that connects to a MySQL database, processes a form submission, and displays the results. The form has fields for name, nim, jenis kelamin, and fakultas. The code uses htmlspecialchars to escape the input data. The form is styled with Bootstrap classes. The code is as follows:

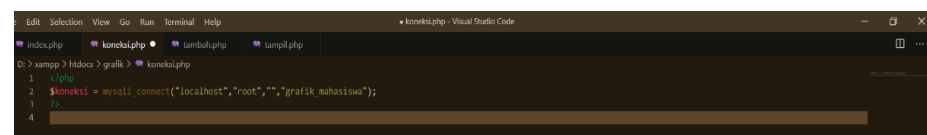
```
1 <?php include("koneksi.php");
2
3 <center><font size="6">tambah Data</font></center>
4 <br>
5 <div>
6     <?php
7     if(isset($_POST['submit'])){
8         $nama      = $_POST['nama'];
9         $nim       = $_POST['nim'];
10        $jenis_kelamin = $_POST['jenis_kelamin'];
11        $fakultas   = $_POST['fakultas'];
12
13        $sql = mysqli_query($koneksi,"INSERT INTO mahasiswa VALUES ('','$nama','$nim','$jenis_kelamin','$fakultas')"); or die(mysqli_error($koneksi));
14
15        if($sql){
16            <script>alert("Berhasil menambahkan data."); document.location="tampil.php";</script>;
17        }else{
18            <div class="alert alert-warning">Gagal melakukan proses tambah data.</div>;
19        }
20    }
21 </div>
22
23 <form action="" method="post">
24     <div class="item form-group">
25         <label class="Hamac/label">
26             <div class="">
27                 <input type="text" name="nama" class="form-control" size="50" required>
28             </div>
29         </div>
30         <div class="">
31             <label class="">nim</label>
32             <div class="">
33                 <input type="text" name="nim" class="" size="50" required>
34             </div>
35         </div>
36         <div class="item form-group">
37             <label class="">jenis kelamin</label>
38             <div class="">
39                 <input type="text" name="jenis_kelamin" class="form-control" size="50" required>
```

Gambar 2.1 index.php

A screenshot of a Visual Studio Code editor window showing the code for index.php. The code includes a PHP script that connects to a MySQL database, processes a form submission, and displays the results. The form has fields for name, nim, jenis kelamin, and fakultas. The code uses htmlspecialchars to escape the input data. The form is styled with Bootstrap classes. The code is as follows:

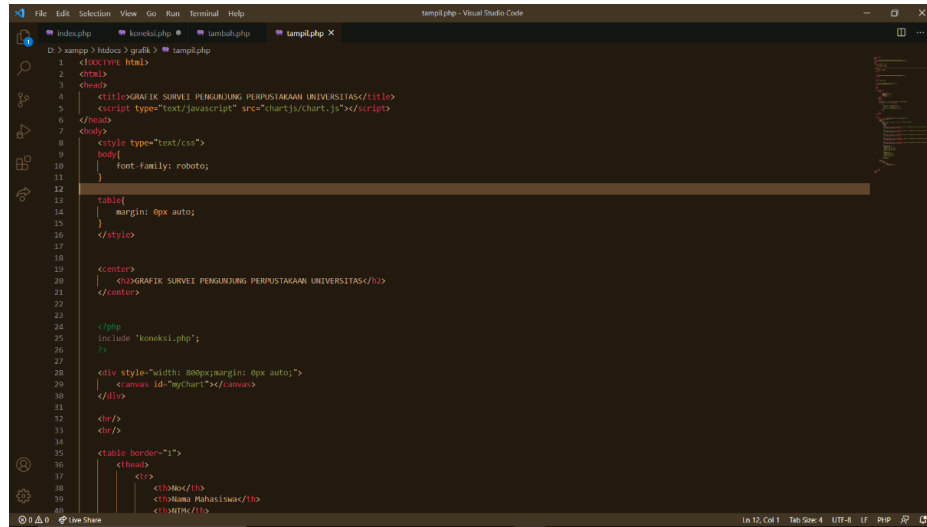
```
40 </div>
41 </div>
42 <div class="item form-group">
43     <label class="">fakultas</label>
44     <div class="">
45         <input type="text" name="fakultas" class="form-control" size="50" required>
46     </div>
47 </div>
48 <div class="item form-group">
49     <div class="">
50         <input type="submit" name="submit" class="btn btn-primary" value="tambah">
51     </div>
52 </div>
53 </form>
54 </div>
```

Gambar 2.2 index.php

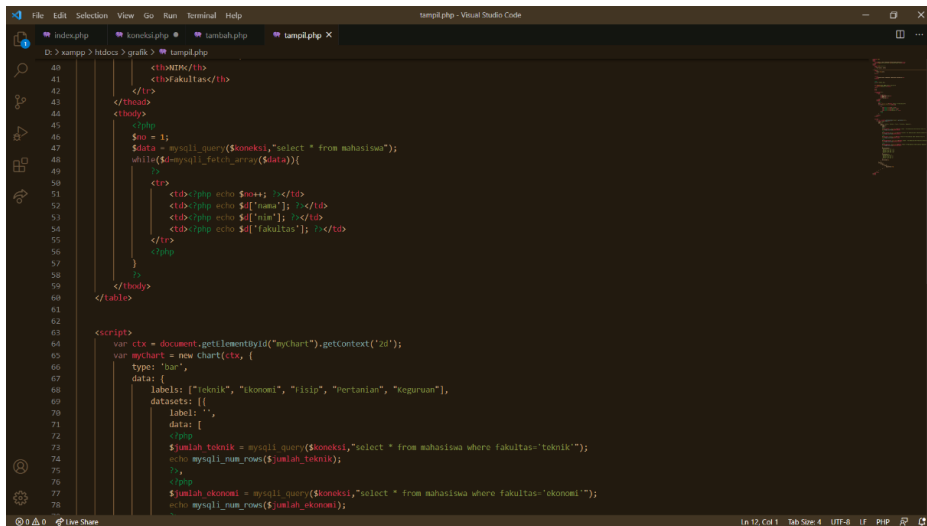
A screenshot of a Visual Studio Code editor window showing the code for koneksi.php. The code includes a PHP script that connects to a MySQL database. The code is as follows:

```
1 <?php
2 $koneksi = mysqli_connect("localhost","root","","grafik_mahasiswa");
3
4
```

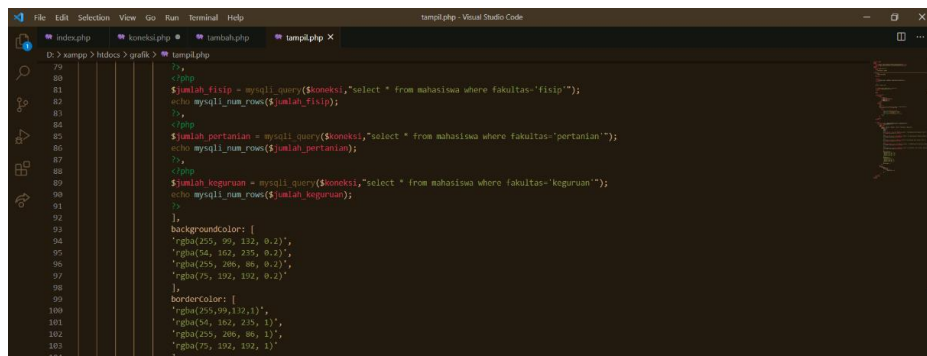
Gambar 2.3 koneksi.php



Gambar 2.4 tampil.php



Gambar 2.5 tampil.php



Gambar 2.6 tampil.php

```

File Edit Selection View Go Run Terminal Help
tampil.php - Visual Studio Code
index.php  koneksi.php  tambah.php  tampil.php X
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
    },
    options: {
      scales: {
        yAxes: [{
          ticks: {
            beginAtZero: true
          }
        }]
      }
    }
  }
}
</script>
</body>
</html>

```

Gambar 2.7 tampil.php

Tambah Data

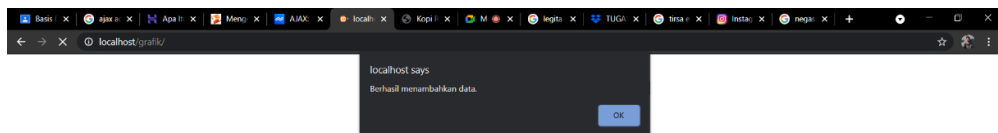
Nama
Jonathan Oktavianus

NIM
193030203322

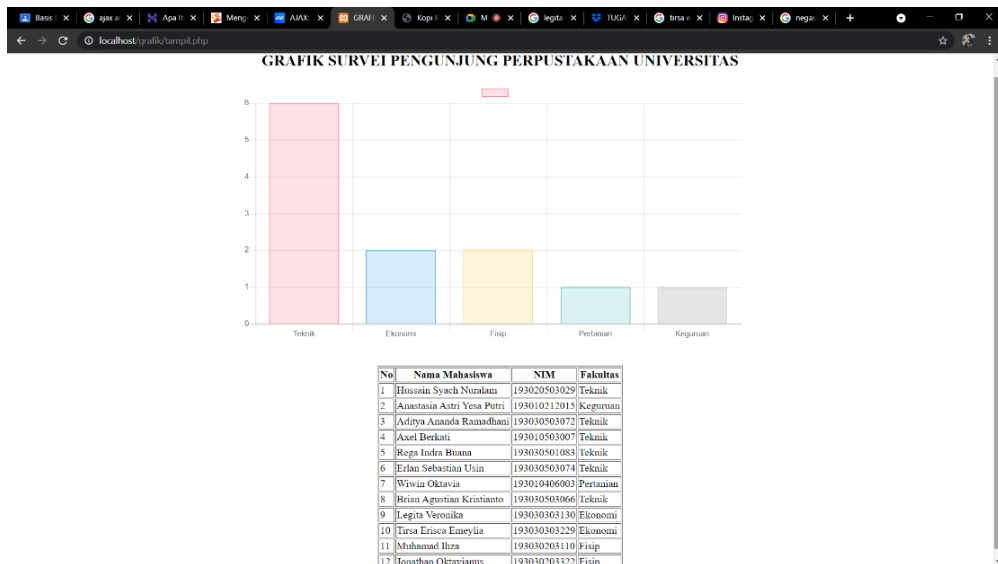
Jenis kelamin
laki laki

Fakultas
Fiisip

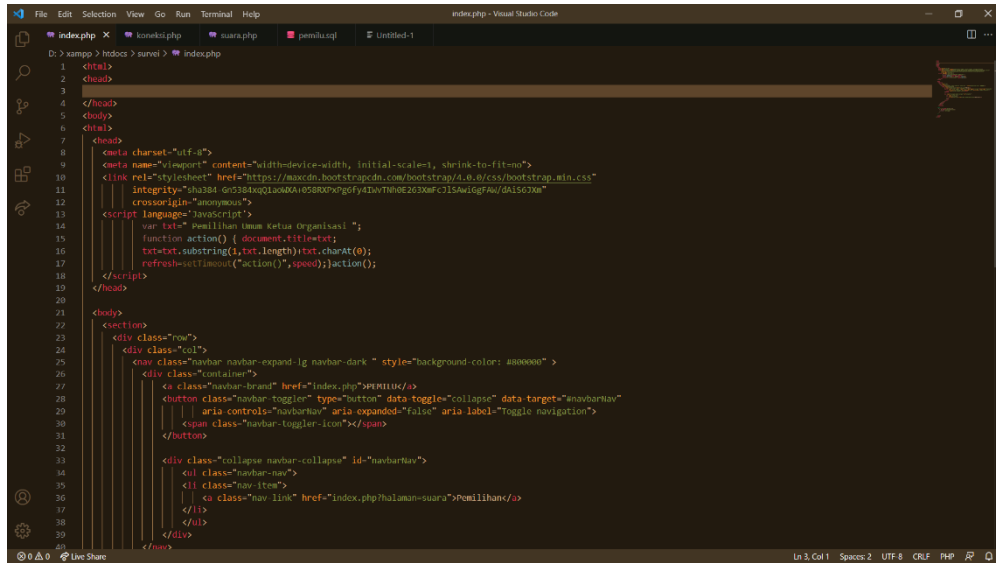
Gambar 2.8 Laman Tambah Data



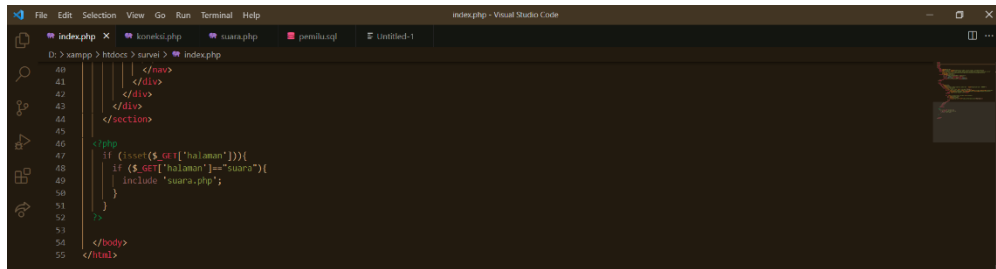
Gambar 2.9 Data Berhasil Ditambahkan



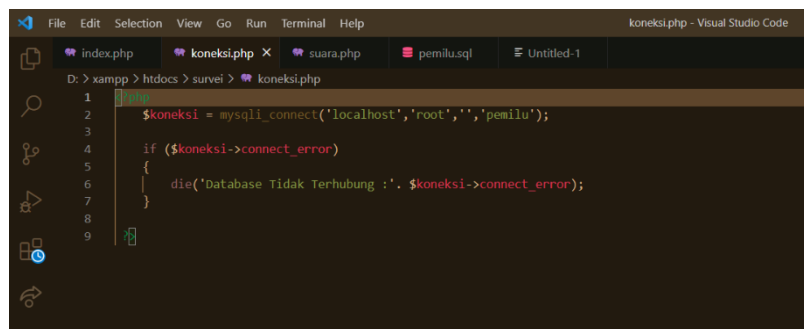
Gambar 2.10 Grafik Pengunjung Perpustakaan

A screenshot of a Visual Studio Code editor window showing the code for index.php. The file explorer on the left shows a project structure with files: index.php, koneksi.php, suara.php, pemilu.sql, and Untitled-1. The code in index.php includes a meta charset, viewport, and a Bootstrap CSS link. It also contains a JavaScript function for a refresh timer. The main content is a Bootstrap navigation bar with a toggle button and a list of links, including one to 'index.php?halaman=suara' for 'Pemilihanc'.

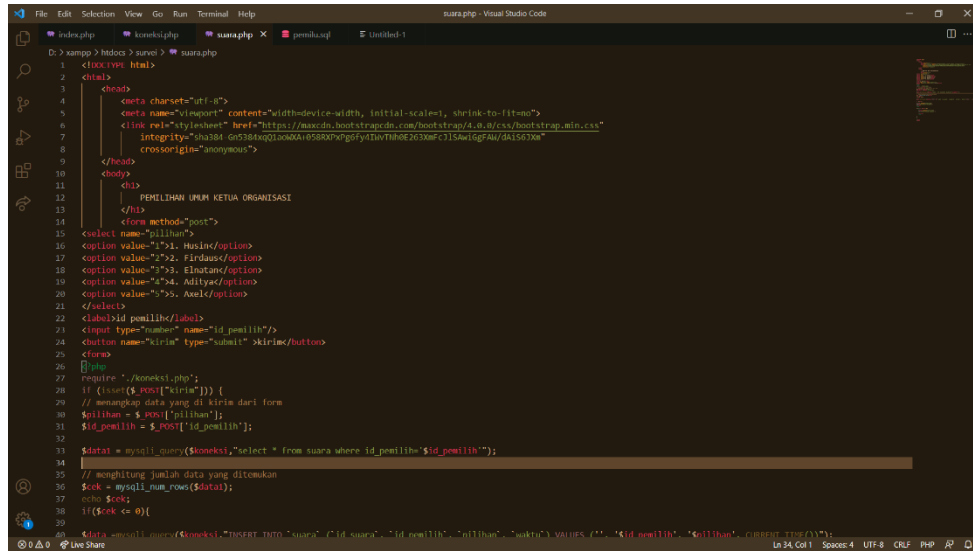
Gambar 2.11 index.php

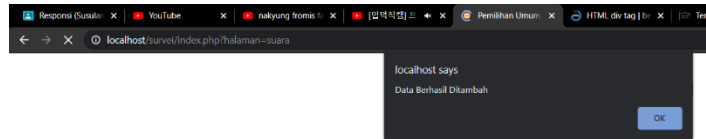
A screenshot of a Visual Studio Code editor window showing the code for index.php. The file explorer on the left shows a project structure with files: index.php, koneksi.php, suara.php, pemilu.sql, and Untitled-1. The code in index.php includes a meta charset, viewport, and a Bootstrap CSS link. It also contains a JavaScript function for a refresh timer. The main content is a Bootstrap navigation bar with a toggle button and a list of links, including one to 'index.php?halaman=suara' for 'Pemilihanc'.

Gambar 2.12 index.php

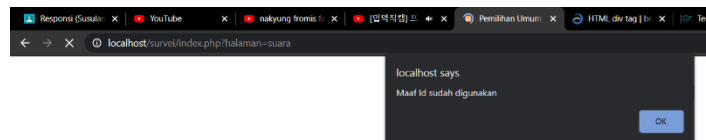
A screenshot of a Visual Studio Code editor window showing the code for koneksi.php. The file explorer on the left shows a project structure with files: index.php, koneksi.php, suara.php, pemilu.sql, and Untitled-1. The code in koneksi.php includes a MySQL connection function that takes 'localhost', 'root', and 'pemilu' as parameters. It checks for connection errors and displays a message if the database is not connected.

Gambar 2.13 koneksi.php





Gambar 2.17 Web Voting



Gambar 2.18 Web Voting